



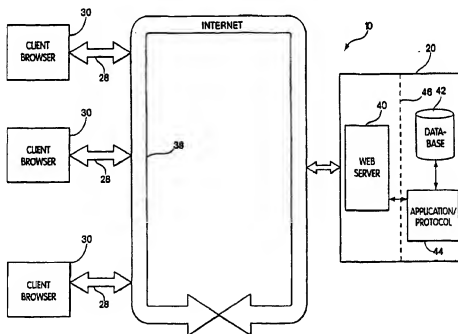
INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 : H04L 29/00		A2	(11) International Publication Number: WO 00/57612
			(43) International Publication Date: 28 September 2000 (28.09.00)
(21) International Application Number: PCT/US00/07529		(81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).	
(22) International Filing Date: 22 March 2000 (22.03.00)			
(30) Priority Data: 60/125,468 22 March 1999 (22.03.99) US			
(71) Applicant: WEBXI [US/US]; 83 Cambridge Street, Burlington, MA 01803 (US).			
(72) Inventor: MCCARTHY, James; 27 Village Street, Marblehead, MA 01945 (US).			
(74) Agents: MAZZARESE, Robert, A. et al.; Foley, Hoag & Eliot, LLP, One Post Office Square, Boston, MA 02109 (US).			

Published

Without international search report and to be republished upon receipt of that report.

(54) Title: HIGH SPEED STREAMING PROTOCOL



(57) Abstract

Provided herein is a high speed streaming protocol for improving the efficiency of application-to-application communication over networks, including Wide Area Networks (WAN). The high speed streaming protocol uses a request-dispatch mechanism which allows an application to embed multiple server-side function requests without waiting for a confirmation for each function call.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SE	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakhstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

HIGH SPEED STREAMING PROTOCOL

Background of The Invention

1. Field of the Invention

This application relates to the field of information technology and more particularly to the field of communication protocols.

2. Description of Related Art

Computer technology has revolutionized information delivery, storage, retrieval and manipulation. In particular, the computer network offers the possibility of improved systems for offering information to geographically distant users. By linking together several computers and by providing shared resources and cross-platform communications, the computer network provides improved access to sophisticated information and applications by users at remote locations.

The present invention may be better understood by reference to a number of commonly used terms from the field of computer networks, definitions of which are as follows:

The term "client," as used herein, encompasses any data processing systems suitable for operating a processor and for establishing a communication link, such as a link to an Internet site. An Internet site can be any program running on a data processing platform that connects to the Internet and that receives access requests, whether under HTTP, FTP or any other conventional or proprietary transfer protocol.

The term "application program," as used herein, encompasses any computer file that contains data in a format for being accessed and processed by the processing unit of a computer.

5 The term "disk," as used herein, encompasses any memory device that can store computer data and that provides an interface for accessing the stored data.

The term "network," as used herein, encompasses any system comprising a series of computers linked by telecommunications networks and may include the Internet, intranets, or other computer networks.

The term "Internet" means the largest global computer communications network.

10 The term "World Wide Web" means a large global computer communications network that comprises a significant part of the Internet.

The term "server," as used herein, encompasses any data processing system on which application programs and Internet sites may be stored for access and processing by client computers.

15 The term "web browser," as used herein, encompasses any application program which allows for multimedia presentation of information, including text, images, sound and video clips. A web browser allows a user to connect by the Internet to different sites on the Internet.

The term "HTTP" as used herein, shall encompass the "HyperText Transfer Protocol", which shall mean a protocol under which messages are sent over the Internet from client computers to server computers in the client-server model of distributed computing.

5 One of the most widely accepted and heavily used networks is the Internet. The Internet is a global system of interconnected computer networks formed into a single world wide network. A user, through the Internet, can interactively transmit messages with users in different countries. Similarly, a user in the U.S. connected to files and libraries and other jurisdictions such as Europe and Asia, can download files for
10 personal use. Accordingly, the Internet computer network both provides strong communications functions and acts like a universal library, providing electronic access to resources and information available from Internet sites throughout the world. In addition to downloading information, remote user can run applications, through application-to-application communications.

15 One problem with application-to-application communications is that such communications are often slow. This is particularly true in the case of certain relational database applications in which a large number of users remotely access large databases in a large number of transactions. Accordingly, methods and systems are needed to improve network performance in order to speed the operation of remote applications
20 using networks, particularly relational database applications.

Summary Of The Invention

Provided herein is a high speed streaming protocol for improving the efficiency of application-to-application communication over networks, including Wide Area Networks (WAN). The high speed streaming protocol uses a request-dispatch mechanism which allows an application to embed multiple server-side function requests
5 without waiting for a confirmation for each function call.

Brief Description Of Drawings

FIG. 1 is a schematic diagram depicting the elements of an embodiment of the present invention.

Detailed Description of the Preferred Embodiment(s)

10 Referring to FIG. 1, the present invention includes a system 10, that comprises a host 20 and a client 30. The client 30 may be any conventional device for receiving, storing, manipulating, retrieving and transmitting data, such as a conventional computer. Thus, the client 30 may include a central processing unit, as well as other conventional elements, such as memory, a display, and a keyboard. The client 30 may include a
15 device for providing an network connection 28, such as a modem. Any other form of network connection 28 may be used. The network connection may connect the client 30 to a communications network 38, such as the Internet. In an embodiment, as depicted in Figure 1, the client 30 is a web browser.

Also connected to the network 38 may be the host 20, which may include a
20 server 40, which may be a conventional server, such as a web server, configured to handle network communications, such as Internet communications using the TCP/IP

protocol. The host 20 may contain files that include data, applications and other content that can be accessed over the network 38 by the client 30. Applications running on the client 30 can then access and run applications running on the host 20, permitting the client 30 to retrieve, store and manipulate the information obtained over the network 38 from a remote location.

The applications running on the host 20 may include a database 42, which in an embodiment may be a relational database. The database 42 may sit behind a firewall 46 or similar security mechanism. The web server 40 may relay queries to the database 42, which may be parsed and handled in a conventional fashion, and the web server 40 may take results from the database 42 and relay them back, via the network 38 to the client 30 that initiated the query. An interface 44 may be provided between the web server 40 and the database 42. The interface 44 may take a variety of configurations, ranging from a conventional server-database link as used in conventional systems that permit access to databases over the Internet. It should be understood that the interface 44 could be a separate unit or could be included as part of either the web server 40 or the database 42. The interface 44 may constitute an application program, sometimes known as "middleware" running between the web server 40 and the database 42. It should be understood that the web server 40 could be one of multiple web servers accessing the same database 42; thus, the database 42 need not reside on the same machine as the web server 40.

In situations in which a large number of users wish to use clients 30 to access a high volume of data contained in a relational database 42, the speed of the system may

be significantly compromised. Accordingly, protocols for improving system speed for such uses are desired. Such a protocol can be included or embodied in the interface 44.

The present disclosure provides a High Speed Streaming Protocol ("HSSP") embodied in the interface 44, which in an embodiment may be the WebXi version 1.0

5 High Speed Streaming Protocol.

Most application-to-application communication is performed using a standard Remote Procedure Call (RPC) mechanism. RPC protocols are based on a request/response calling method, which results in multiple packets for each server-based procedure invocation. The number of network packets needed to carry the procedure

10 calls is directly proportional to the number of remote procedure calls that are made. Most of the current database client/server interfaces are based on an application-specific RPC mechanism that follows the request/response architecture. The following is an example of the network packets required to implement a conventional request/response architecture:

15 BEGIN LOGICAL UNIT of WORK

SEND: Request1,Parameter1,Parameter2

RECEIVE: Request1,Confirmation

SEND: Request2,Parameter1,Parameter2

RECEIVE: Request2,Confirmation

20

SEND: Request10,Parameter1,Parameter2

RECEIVE: Request10,Confirmation,Response Data

END LOGICAL UNIT of WORK

WebXi HSSP is a unique communication protocol designed to improve the efficiency of application-to-application communication over networks, including Wide Area
5 Networks (WAN). The objective of HSSP is to increase network performance and improve the reliability of server-based applications. WebXi HSSP uses a request-dispatch mechanism which allows an application to embed multiple server-side function requests without waiting for a confirmation for each function call.

HSSP is a core component which may used in the interface 44 to improve the
10 efficiency of database requests over WAN for WebXi products. The following is a description of the HSSP protocol and how it differs from conventional application-to-application communication protocols.

An objective of HSSP is to increase network performance. The WebXi HSSP is
15 based on a client-directed request protocol that increases performance by significantly reducing the number of network packets required to accomplish a logical unit of work. HSSP is designed to minimize the number of network packets by allowing the requesting application to determine the end point of a request. The body of the request can contain multiple server based procedure calls that will build a response packet
20 encapsulating all of the procedure calls up until the end point defined by the requesting application. This allows the requesting application to send a single network packet as logical units of work rather than multiple network packets for each procedure call in a

logical unit of work. The following is an example of the network packets required to implement the HSSP client-directed request protocol:

BEGIN LOGICAL UNIT of WORK

SEND: Request1,Parameter1,Parameter2,Request2,Parameter1,

5 Parameter2 ... Request10,Parameter1,Parameter2,ENDREQ

RECEIVE: Request10,Response Data

END LOGICAL UNIT of WORK

In the above HSSP example the requesting application made ten separate procedure calls with only two network packets, while the RPC-based example would usually require up to twenty network packets to perform the same unit of work. The requesting application was only concerned about the response data from the final procedure call, so HSSP used an implied confirmation for each procedure call when the call did not have any response data. All procedures assume a successful completion unless an error occurs and a failure is returned, ending the logical unit of work at the point of failure. When a failure occurs, the requesting application and the server-based application are responsible for handling the failure accordingly.

HSSP may be applied as a database request interface for the interface 44. As a result, the number of network packets required to execute a database request is usually reduced from thirty or more network packets to less than five. This can result in significant performance improvements for an application over wide area networks.

A second objective of HSSP is to improve the reliability of server-based procedure calls. HSSP may be based on an internal data packet description that improves

the reliability of the server-based application by embedding a description of each data element of the packet. The data packet description uses a single byte immediately preceding each atomic data type to allow the server based application to query the data stream and take action based on the next data type to be processed in the stream. The

5 HSSP data packet description allows the server-based application to recover from unanticipated data types that may be encountered in the data stream when an application changes versions. The server-based application can choose to proceed with an anticipated condition or gracefully return a failure when this unanticipated condition occurs. The HSSP data packet description improves the reliability of a server-based

10 application by eliminating blind data streaming which is usually present in most RPC mechanisms.

The internal description of the HSSP packet forms a strong barrier of protection from unanticipated conditions and provides for an automatic or application handled mechanism of recovery. For example, the HSSP dispatch mechanism uses the data

15 packet description to automatically skip to the next operation when a dispatched function returns. Any data that is not handled by a procedure would be ignored and the next dispatched procedure would be called.

The High Speed Streaming Protocol combines the internal data packet description along with client-directed logical units of work to achieve a reliable

20 application-to-application streaming protocol that improves network performance. The following detailed diagram illustrates the combined description that makes up the High Speed Streaming Protocol.

Packet Header:	TOTAL LENGTH	PACKET VERSION				
Packet Body:	TYPE	OPCODE	TYPE	DATA	TYPE	OPCODE
Packet End:	TYPE	OPEND				

Packet header precedes the start of every network packet. The packet header is used to determine the total length to read from the network before dispatching the packet and the stream version to be used for packet interpretation. The length value is used to prevent overrunning the read buffer and should not exceed the negotiated packet maximum. The version is used as a synchronizing mechanism to allow each application to determine the expected packet version for procedure interpretation. Each communicating application transmits their local packet version when sending or responding to a request. This allows each application to determine the remote version for each procedure call and respond accordingly. The application can build in backward compatibility based on the remote packet version.

The packet body is made up of a series of data type codes and data elements that are application specific. The application-specific packet body is made up of a series of procedure operation codes that precede the optional procedure parameters. The stream dispatcher uses the procedure operation code data type to determine the application specific code module to call in order to handle the procedure parameters. When the code module is called it will stream the procedure parameters out of the network packet. When the procedure returns the stream dispatcher will move to the next procedure operation code to determine the next action in the stream.

The packet end is a special procedure indicator that indicates to the server that the requesting application is now waiting for a response. The stream dispatcher will assume that any preceding procedure operation codes have built a response packet and will write the current response packet back to the requesting application. If none of the dispatched procedures contain response data the end operation code will simply be echoed back otherwise, the response data along with the end operation code are returned. The end operation code is sent in both directions so that each end of the communication stream knows when the other end is sending or waiting. The requesting application will send a close operation code when it has completed all requests and wants to terminate the connection.

Both the requesting application and the server have the same base code for streaming and dispatching requests. As a result, when the server receives a request it will reply with a stream that is formatted exactly the same way as the requesting application transmits it. Also, with knowledge of the packet version, the server is able to format the packet body in a manner that is backward compatible with the requesting application.

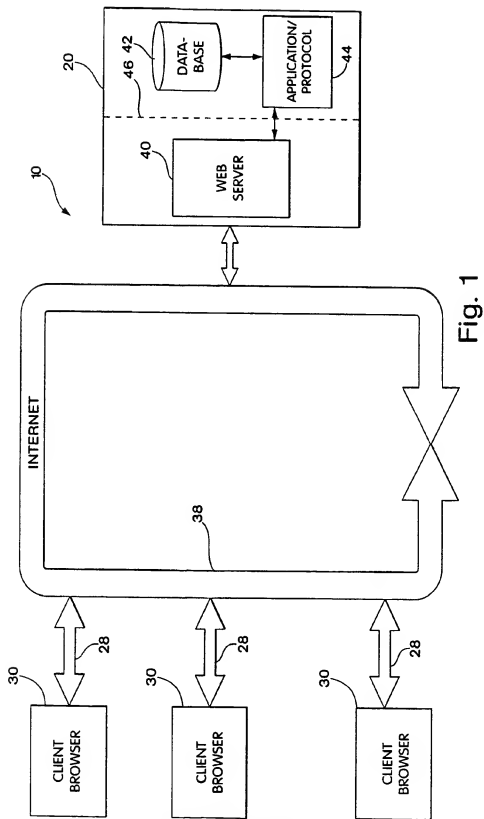
While the invention has been disclosed in connection with the preferred embodiments shown and described in detail, various modifications and improvements thereon will become readily apparent to those skilled in the art. Accordingly, the spirit and scope of the present invention is to be limited only by the following claims.

Claim(s)

What is claimed is:

1. A method of application-to-application communication in a computer network having a client application and a server application comprising:
providing a request-dispatch mechanism of the server application for handling requests of the client application to the server application; and
allowing an application to embed multiple server-side function requests without waiting for a confirmation for each function call.

1/1



SUBSTITUTE SHEET (RULE 26)

THIS PAGE BLANK (USPTO)

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
28 September 2000 (28.09.2000)

PCT

(10) International Publication Number
WO 00/57612 A3

- (51) International Patent Classification: H04L 29/06 (81) Designated States (national): AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.
- (21) International Application Number: PCT/US00/07529
- (22) International Filing Date: 22 March 2000 (22.03.2000)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data: 60/125,468 22 March 1999 (22.03.1999) US
- (71) Applicant: WEBXI [US/US]; 83 Cambridge Street, Burlington, MA 01803 (US).
- (72) Inventor: MCCARTHY, James; 27 Village Street, Marblehead, MA 01945 (US).
- (74) Agents: MAZZARESE, Robert, A. et al.; Foley, Hoag & Eliot, LLP, One Post Office Square, Boston, MA 02109 (US).
- (84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

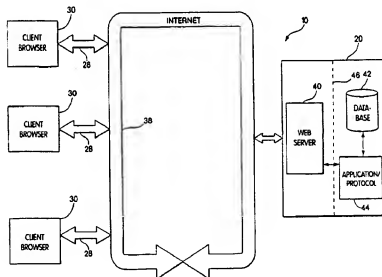
Published:

- With international search report
- Before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments.

(88) Date of publication of the international search report:
18 January 2001

[Continued on next page]

(54) Title: APPLICATION LAYER PROTOCOL



(57) Abstract: Provided herein is a high speed streaming protocol for improving the efficiency of application-to-application communication over networks, including Wide Area Networks (WAN). The high speed streaming protocol uses a request-dispatch mechanism which allows an application to embed multiple server-side function requests without waiting for a confirmation for each function call.



For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

International Application No
PCT/US 00/07529

2

INTERNATIONAL SEARCH REPORT

International Application No.

PCT/US 00/07529

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 859 853 A (CARLSON DAVID GLENN) 12 January 1999 (1999-01-12) abstract	1
X	US 5 325 361 A (LEDERER JEFFREY H ET AL) 28 June 1994 (1994-06-28) abstract column 2, line 8 - line 44 figures 5-7	1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No.

PCT/US 00/07529

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5859853 A	12-01-1999	EP 0909490 A	21-04-1999
		JP 11511940 T	12-10-1999
		WO 9749209 A	24-12-1997
US 5325361 A	28-06-1994	AU 5672294 A	22-06-1994
		EP 0628230 A	14-12-1994
		WO 9413086 A	09-06-1994

Form PCT/ISA210 (patent family annex) (July 1992)

THIS PAGE BLANK (USPTO)